



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/670,703	09/25/2003	John D. Irish	ROC920030090US1	6165
30206	7590	01/25/2010	EXAMINER	
IBM CORPORATION ROCHESTER IP LAW DEPT. 917 3605 HIGHWAY 52 NORTH ROCHESTER, MN 55901-7829			ROJAS, MIDYS	
			ART UNIT	PAPER NUMBER
			2185	
			NOTIFICATION DATE	DELIVERY MODE
			01/25/2010	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

rociplaw@us.ibm.com

Office Action Summary	Application No. 10/670,703	Applicant(s) IRISH ET AL.	
	Examiner MIDYS ROJAS	Art Unit 2185	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 14 September 2009.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-34 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-34 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

Response to Arguments

1. Applicant's arguments, filed 9/14/2009, with respect to the rejection(s) of claim(s) 1-34 under 35 USC 103 have been fully considered but are not persuasive.

In pages 12-15 of the remarks, Applicant argues that the references being relied upon do not teach "determining whether a free group entry of a size required by a portion of the set of data exists in memory pre-allocated with a group size in one of a plurality of sections of a memory". Specifically, in page 14, Applicant argues that Bonola does not teach whether a free **group entry** of a size required by a portion of the set of data exists in memory **pre-allocated with a group size** in one of a plurality of sections of memory and that the sub regions of Bonola are not planned in advance and therefore, cannot be pre-allocated with a size.

The examiner does not agree.

Bonola discloses determining whether a free group entry of a size required by a portion of the set of data exists in a memory (an application program executes HeapAlloc instruction to allocate a heap sub region of size X bytes, paragraph 0036. **The heap sub region of size X represents the required group size entry.** In executing HeapAlloc, the processor examines the heap data structure to determine if there is an appropriately sized heap sub-region, of size X bytes as required, in the free list wherein the free list contains unassigned heap sub-regions, paragraph 0037) pre-allocated with a group size in one of a plurality of sections of a memory (**pre-allocation of heap and heap sub-regions in the execution of HeapCreate(n), paragraph 0030**

wherein pre-allocating a heap, the memory is pre-allocating a section of the memory with a particular group size equivalent to the size of the heap). The required size of X bytes represents the free group entry of the size required by a portion of the data, as claimed; and the pre-allocated memory is represented by the creation of the heap, wherein memory is pre-allocated to the heap from memory.

In pages 15-17 of the remarks, Applicant argues that the references being relied upon do not teach that "if the memory includes one or more sections of an unallocated size, allocating one of the sections of an unallocated size to the size required by the portion of the set of data thereby creating a section of a dynamically allocated size, the section of the dynamically allocated size including one or more group entries of the size required by the portion of the set of data, the dynamically allocated size being the smallest sized group entry necessary to store the portion of the set of data". Specifically, in page 17, Applicant argues that no disclosure whatsoever in Shaylor or Bonola teach group entries in the available physical address space nor that the available space includes one or more group entries.

The examiner does not agree.

Shaylor teaches determining whether the memory includes one or more sections of an unallocated size (if a task requests additional memory space for the heap, it is determined if the address space immediately adjacent to the heap is available, paragraph 0038; wherein requesting additional memory space for the heap, the system has determined that a group entry of the size required by the portion of the set of data

Art Unit: 2185

does not exist in the memory); and if the memory includes one or more sections of an unallocated size (if the space adjacent to the heap is available...), allocating one of the sections of an unallocated size to the size required by the portion of the set of data thereby creating a section of a dynamically allocated size (allocate the requested memory address space from the space adjacent to the heap, paragraph 0038), the section of the dynamically allocated size including one or more group entries of the size required by the portion of the set of data (**the allocated space is equivalent to the additional requested memory address space, thus including one group entry of the size required by the data**, see paragraph 0038), the dynamically allocated size being the smallest sized group entry necessary to store the portion of the set of data (the allocated space is equivalent to the requested memory address space, therefore, it is the smallest size necessary, paragraph 0038).

In pages 17-18 of the remarks, Applicant argues that the references being relied upon do not teach the newly added limitation: "adapted to allocated memory into a section including one or more group entries".

The examiner does not agree.

Bonola discloses a method of allocating memory into a section including one or more group entries (a processor allocating a portion of memory 119 called a heap by executing HeapCreate(n) and setting aside a continuous number of bytes for the creation of the heap, paragraph 0029; wherein a heap may be divided into heap sub-

portions that can be individually assigned to applications, paragraph 0030. The heap sub-portions represent the group entries).

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

3. Claims 1-34 are rejected under 35 U.S.C. 103(a) as being obvious over Bonola (2001/0011338) in view of Shaylor (US 2002/0108025 A1).

Regarding Claim 1, Bonola discloses a method of allocating memory into a section including one or more group entries (a processor allocating a portion of memory 119 called a heap by executing HeapCreate(n) and setting aside a continuous number of bytes for the creation of the heap, paragraph 0029; wherein a heap may be divided into heap sub-portions that can be individually assigned to applications, paragraph 0030. The heap sub-portions represent the group entries), the method comprising:

receiving a set of data (in the process of executing an instruction within an application program, paragraph 0013);

determining whether a free group entry of a size required by a portion of the set of data exists in a memory; an application program executes HeapAlloc instruction to allocate a heap sub region of size X bytes, paragraph 0036. In executing HeapAlloc, the processor examines the heap data structure to determine if there is an appropriately

sized heap sub-region in the free list wherein the free list contains unassigned heap sub-regions, paragraph 0037) pre-allocated with a group size in one of a plurality of sections of a memory (pre-allocation of heap and heap sub-regions in the execution of HeapCreate(n), paragraph 0030 wherein pre-allocating a heap, the memory is pre-allocating a section of the memory with a particular group size equivalent to the size of the heap);

if a free group entry of the size required by the portion of the set of data does not exist in one of the plurality of sections of the memory (if N exceeds M, then there is no heap sub-region within the heap that can accommodate the demand..., paragraph 0038), determining whether the memory includes one or more sections of an unallocated size (unassigned heap sub-region, paragraph 0034-0035).

Bonola does not teach that if the memory includes one or more sections of an unallocated size, allocating one of the sections of an unallocated size to the size required by the portion of the set of data thereby creating a section of a dynamically allocated size, the section of the dynamically allocated size including one or more group entries of the size required by the portion of the set of data, the dynamically allocated size being the smallest sized group entry necessary to store the portion of the set of data.

Shaylor teaches determining whether the memory includes one or more sections of an unallocated size (if a task requests additional memory space for the heap, it is determined if the address space immediately adjacent to the heap is available, paragraph 0038; wherein requesting additional memory space for the heap, the system

has determined that a group entry of the size required by the portion of the set of data does not exist in the memory); and if the memory includes one or more sections of an unallocated size (if the space adjacent to the heap is available...), allocating one of the sections of an unallocated size to the size required by the portion of the set of data thereby creating a section of a dynamically allocated size (allocate the requested memory address space from the space adjacent to the heap, paragraph 0038), the section of the dynamically allocated size including one or more group entries of the size required by the portion of the set of data (the allocated space is equivalent to the additional requested memory address space, thus including one group entry of the size required by the data, see paragraph 0038), the dynamically allocated size being the smallest sized group entry necessary to store the portion of the set of data (the allocated space is equivalent to the requested memory address space, therefore, it is the smallest size necessary, paragraph 0038). It would have been obvious to one of ordinary skill in the art that time the invention was made to modify the invention of Bonola to include the heap memory allocation method of Shaylor since this allows for the dynamic allocation of memory to the heap as needed (paragraph 0006).

Regarding Claim 2, Bonola discloses a method wherein determining whether a free group entry of the size required by the portion of the set of data exists in one of a plurality of sections of the memory includes determining whether a free group entry of the size required by the portion of the set of data for uniquely identifying each portion of the set of data exists in one of the plurality of sections of memory (...no heap sub-region

within heap 302 can accommodate a demand for X bytes of memory from application 202, paragraph 0038 wherein the memory is allocated to the heap data structure).

Regarding Claim 3, Bonola discloses a method wherein determining whether the memory includes one or more sections of an unallocated size includes accessing a control structure for one or more sections of the memory, the control structure storing information about the structure of a section (heap data structure 304).

Regarding Claim 4, Bonola discloses a method further comprising, from the section of a dynamically allocated size, allocating an initial group entry of the size required by the portion of the set of data for storing the portion of the set of data (... the address of the heap sub-region contained in the entry is then assigned to the application, paragraph 0037).

Regarding Claim 5, Bonola discloses a method further comprising: receiving a modified set of data; determining whether a portion of the modified set of data may be stored more efficiently in a group entry of a different size from another section of the memory such that the aggregate number of unused entries in the group entries used for storing the modified set of data is minimized (...if the remaining half of the heap sub-region contains twice as much memory as is required by the requesting application, the remaining half of the heap sub-region can be further divided..., paragraph 0039); allocating a group entry of the different size required by the portion of the modified set of

data from another section of the memory to store the portion of the modified set of data (repeating steps 532-540 until a heap sub-region is yielded which does not have twice as much memory as is required and then assigning the sub-region to the application... paragraph 0039); and de-allocating the initial group entry to the section of memory from which the initial group entry was allocated (wherein de-allocation of the initial group entry occurs when the newly selected heap sub-region is placed in the N list and replaces the initial entry, Figure 5).

Regarding Claim 6, Bonola discloses a method further comprising updating the control structure that stores information about the structure of the other section (...an entry is removed from the N list in the free list 306, paragraph 0038).

Regarding Claim 7, Bonola discloses a method further comprising updating the control structure that stores information about the structure of the section of memory from which the initial group entry was allocated (...an entry is removed from the N list in the free list 306, paragraph 0038).

Regarding Claim 8, Bonola discloses a method wherein de-allocating the initial group entry to the section of memory from which the initial group entry was allocated leaves all entries of the section unused (when the application is finished using an assigned heap sub-region, it can return the heap sub-region to the heap using the HeapFree instruction, thus leaving those entries unused, paragraph 0040).

Regarding Claim 9, Bonola discloses a method further comprising clearing the group entry size allocation of the section (HeapFree instruction clears the allocation of the sections since the heap contains unassigned memory, paragraph 0040-0041).

Regarding Claim 10, Bonola discloses a method further comprising, if the memory does not include one or more sections of an unallocated size, determining whether a free group entry of a size larger than the size required by the portion of the data exists (Paragraph 0039, heap sub-region with the higher memory address is entered into the N list...), wherein sections allocated to the smallest available size larger than the size required by the portion of the data are checked prior to sections allocated to larger available sizes (continue dividing sub-region in half until sub-region contains less than twice as much memory as required).

Regarding Claim 11, Bonola discloses a method further comprising, if a free group entry of a size larger than the size required by the portion of the data exists in a section allocated to a size larger than the size required by the portion of data, allocating an initial group entry of the size larger than the size required by the portion of the set of data from the section allocated to a size larger than the size required by the portion of the data for storing the portion of the set of data (paragraph 0039, if the sub-region contains less than twice as much memory as needed... the sub-region should be assigned).

Regarding Claim 12, Bonola discloses a method further comprising, if a free group entry of a size larger than the size required by the portion of the data does not exist outputting an error condition (Paragraph 0038, there is no sub-region within the heap that can accommodate a demand for X bytes... HeapAlloc instruction will fail...).

Claim 13 is rejected using the same rationale as that of Claim 5.

Claim 14 is rejected using the same rationale as that of Claim 6.

Claim 15 is rejected using the same rationale as that of Claim 7.

Claim 16 is rejected using the same rationale as that of Claim 8.

Claim 17 is rejected using the same rationale as that of Claim 9.

Claim 18 is rejected using the same rationale as that of Claim 1 wherein the memory is represented by system memory 114, the plurality of registers are within heap data structure 304, and the dynamic allocation logic is equivalent to the method for dynamically allocating memory in a computer system of the invention (paragraph 0013, and Figure 3).

Claim 19 is rejected using the same rationale as that of Claim 2.

Claim 20 is rejected using the same rationale as that of Claim 3.

Claim 21 is rejected using the same rationale as that of Claim 4.

Claim 22 is rejected using the same rationale as that of Claim 5.

Claim 23 is rejected using the same rationale as that of Claim 6.

Claim 24 is rejected using the same rationale as that of Claim 7.

Claim 25 is rejected using the same rationale as that of Claim 8.

Claim 26 is rejected using the same rationale as that of Claim 9.

Claim 27 is rejected using the same rationale as that of Claim 10.

Claim 28 is rejected using the same rationale as that of Claim 11.

Claim 29 is rejected using the same rationale as that of Claim 12.

Claim 30 is rejected using the same rationale as that of Claim 5.

Claim 31 is rejected using the same rationale as that of Claim 6.

Claim 32 is rejected using the same rationale as that of Claim 7.

Claim 33 is rejected using the same rationale as that of Claim 8.

Claim 34 is rejected using the same rationale as that of Claim 9.

Conclusion

4. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

Art Unit: 2185

the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to MIDYS ROJAS whose telephone number is (571)272-4207. The examiner can normally be reached on M-TH 6:00am - 4:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Sanjiv Shah can be reached on (571) 272-4098. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Sanjiv Shah/
Supervisory Patent Examiner, Art Unit 2185

/Midys Rojas/
Examiner, Art Unit 2185

MR